Hospital Patient Management Software Project

Team Webpage: https://github.com/A-Cedeno/Hospital-Database-Project

Authors: Alana Cedeño, Daniel Mathieu, Georenz Koo

Customer: Professor Ingrid Russell

Instructor: Professor Ingrid Russell

1. Introduction

1.1. Purpose

This document is intended to document and outline the requirements for the hospital management software. It is intended for use by members of $teamname as a reference while developing the product and for evaluation of requirements by the client

1.2. Scope

This software is intended for use in a hospital to manage patient data. It will be used by 4 discrete user types with varying access levels and permissions. Computerized patient records will allow for faster access than hard copy records, and will allow for dynamic transformation and access control of that information. This will decrease workload and reduce errors at all stages of the patient care process.

1.3. Definitions acronyms and abbreviations

a. UI - User Interface
b. JDBC - Java database connectivity
c. OS- Operating System
d. JRE - Java Runtime Environment
e. CPU- Central Processing Unit
f. RAM- Random Access Memory
g. ROT-26 - Rotate26, A letter substitution cipher that advances letters by 26 positions

1.4. Overview

The objective of this project is to be able to add and maintain a patient's record, show any lab work done to the patient, and be able to calculate the patient's bill. Furthermore, there will be a login page that will require a username and password to gain access to the system. However, the information that can be accessed will vary by the user's level, whether it is a patient, a nurse, a doctor, or someone from billing.

2. General Description

2.1. Product Perspective

For the moment, this project will be independent given that the team has not begun implementing the software. However, once the program is fully functional and if there is still time, the project might be extended and become part of a larger hospital management system.

2.2. Product functions

The ability to independently access information consistent with a user's permissions as well as the ability for individuals to access, update, and modify information that is stored in the management software.

2.3. User characteristics

a. All users will require basic computer literacy skills, typing, Operating System navigation.
b. **Registration**
    i. The user will need to be able to create new patient entries and populate some initial fields.
    ii. This user will be able to access this product from a typical workstation computer with a keyboard and mouse.
c. **Nurse**
    i. During the initial evaluation, the nurse user will need to be able to access existing patient entries and input additional information including role-specific notes.
    ii. This user may either use a traditional workstation or a portable device such as a laptop or tablet with a touch screen.
d. **Doctor**
    i. This user will need write-access to patient vital information and Nurse notes.
    ii. This user will additionally be able to input treatment information and request tests.
e. **Billing**
    i. This user will need access to administrative information about the patient, billable treatment, and diagnostic procedural line items.
    ii. This user is able to create itemized bills.
    iii. This role does not need access to vitals information.

2.4. General constraints

a. Changes to a patient's record can be done in a save/submit type of form that will be updated in the system.
b. Login-ID and password can only give access to limited information depending on the access level such as patient, nurse, doctor, or billing.

2.5. Assumptions and dependencies

a. For database connections, there is heavy reliance on JDBC drivers, the unavailability of which would greatly change the requirements of the system.
b. This application will be developed with the expectation that Windows OS with Java will be present that will handle all hardware or system calls. The portability of Java applications means that this may be compatible with other OS, however, this configuration is undefined.

3. Specific Requirements

3.1 Functional Requirements

a. Log-In Function
    i. The purpose of a log-in function is to designate different accessibility for individuals depending on their needs.
    ii. A log-in function would have a range of accepted values, normally consisting of an identification value along with a unique and secure passcode.
    iii. The processing that would occur during a log-in would include reading the information provided in the open fields, and verifying the existence of the log-in information within a database or accompanying software. The only abnormal behaviors the function would be likely to experience are logins that cannot be verified by the system.
    iv. When the information has been verified, a version of the UI consistent with that user's permissions will be displayed. If the information cannot be verified, specialized errors will be returned to the user informing them of why the system cannot verify the information they provided.
b. Access Function
    i. The purpose of an access function is to display information to a user that is consistent with their permissions.
    ii. The system will accept values that attempt to navigate the UI, like mouse clicks, keyboard returns, and any other accepted form of navigation.
    iii. The system, either upon log-in or upon receiving a navigation value, will process and format the attributes that can be displayed to a user based on their permissions.

iv. The system will display a version of the UI to the user, and if any error occurs on loading that will be returned to the user, the page will be redirected.

c. Update/Modify Function
   i. The purpose of this function is to allow a user to add or modify values of a certain attribute consistent with their permissions. This would generally include adding a new patient's information to the system or modifying a patient's existing records.
   ii. The range of accepted values would differ for each particular attribute, some may only concern numerical values while others may expect strings of characters.
   iii. To process an update, the system will locate the information that must be accessed, verify that it can be updated at this time depending on the concurrency control implemented, then queue the update and change the information within the system for all users.
   iv. The system will return any errors that may occur while attempting to update a field and will display a visual form of verification if the information was updated successfully.

3.2 External Interface requirements

a. User interface
   i. This application will utilize a Graphical User Interface navigable with mouse and keyboard or touch screen.
   ii. User Interface will be built with Netbeans Swing UI.

b. Hardware interface.
   i. This application can be navigated with input in the form of an operating system compatible pointing device such as a mouse or a touch screen.
   ii. This application will make use of a keyboard for input of information.
   iii. This application will display information on the screen of the computer.

c. Software interface
   i. JDBC Driver/Connector for access to external database.
   ii. Java Runtime Environment.
   iii. IDE: Apache Netbeans IDE.
   iv. MySQL Server.
   v. Communications Interface.
   vi. No network functionality is planned at this time.

3.3 Performance requirements

a. The User Interface must be quick in response time.
b. The system should be able to handle concurrent user access.

3.4 Design Constraints

a. Standards compliance
    i. HIPAA compliance
        1. Access control will be applied to maintain the confidentiality of patient data on a need-to-know basis designated by user type.
        2. Data will be secured with top of the line government spec ROT-26 encryption
b. Hardware limitations
    i. Modest specification general-purpose computer.
    ii. x86 Architecture, 1Ghz CPU. An insufficient processor will result in sluggish performance.
    iii. 8GB RAM. Insufficient RAM will result in program instability.
    iv. 1GB Hard drive space. Insufficient storage will result in the inability to create new records.

3.5 Attributes

a. Availability
    i. The software must be available at all times.
b. Security Requirements
    i. Login ID and Password.
    ii. Changes to information such as records or any valuable data must be synchronized and can only be allowed by a system administrator.
c. Maintainability
    i. Backup data.
    ii. Track errors.